



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/606,547	06/26/2003	James R. Wason	END920020072	5225

7590 09/15/2006

Andrew M. Calderon
Greenblum and Bernstein P.L.C.
1950 Roland Clarke Place
Reston, VA 20191

EXAMINER

NGUYEN, MAIKHANH

ART UNIT PAPER NUMBER

2176

DATE MAILED: 09/15/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

10/606,547

Applicant(s)

WASON, JAMES R.

Examiner

Maikhanh Nguyen

Art Unit

2176

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 05 July 2006.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-50 is/are pending in the application.
- 4a) Of the above claim(s) 39-42 is/are withdrawn from consideration.
- 5) ☐ Claim(s) 16-18 is/are allowed.
- 6) ☒ Claim(s) 1-15, 19-38 and 43-50 is/are rejected.
- 7) ☐ Claim(s) 36-38 is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
 - ☐ Certified copies of the priority documents have been received in Application No. _____.
 - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. This action is responsive to communications: Amendment filed 07/05/2006 to the original application filed 06/26/2003.

Claims 1-38 and 43-50 are currently pending in this application. Claims 1, 12, 16, 24, and 48 have been amended. Claims 39-42 have been withdrawn. Claims 1, 12, 16, 24, and 48 are independent claims.

Applicant is required to cancel non-elected claims 39-42 or other appropriate action (see 37 CFR 1.144) and (MPEP 821.01) in the next response to this office action.

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Claims 1-15, 19, 20, 24-31, 33-35, 43-45, and 47-50 are rejected under 35 U.S.C. 103(a) as being unpatentable over **Prinzing'206** in view of **Prinzing** (U.S. 6,470,364 – issued 10/2002, hereinafter Prinzing'364).

As to claim 1:

Prinzing'206 teaches a method of representing and managing rich text for use by applications (*see the Abstract*), the method comprising the steps of:

- providing one or more classes (*e.g., provides a variety of ... classes*) for use by the applications (*e.g., accessible to GUI application*) to at least create and manage (*e.g., formatting and displaying*) one or more rich text nodes (*e.g., a hierarchy of text elements*) in a memory structure representation representative of rich text (*e.g., storing formatting information and attributes of the text elements and view objects 121 for formatting and displaying text on display device 105*) [see the discussion beginning at col.3, line 43];
- representing the rich text in the memory structure representation (*e.g., text 117 may be organized as a hierarchy of text elements including a document element, paragraphs elements, and character elements; col.3, lines 43-53*); and
- editing the rich text in a document using the memory structure representation to perform editing functions on the document having the rich text as managed and created by the one or more classes (*e.g., allows styles to be added and removed*

modularly by organizing the text within the document and objects that format the text hierarchically...new styles are added by adding the formatting information to a style object and associating a new view object capable of generating a view of the new style with the corresponding text element in the hierarchy; col.2, lines 44-61).

Prinzing'206 does not specifically teach the use of Web based applications and browsers.

Prinzing'364 teaches the use of Web based applications and browsers (*e.g., a hypertext markup language (HTML) editor kit 212...for editing...HTML source code... GUI application including Java source code 222, HTML source code 224; see col.4, line 46-col.5, line 16; see also col.11, lines 31-54).*

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include the feature from Prinzing'364 in the system of Prinzing'206 because it would have provided an improved method for customizing text components that edit a particular type of text and have a particular type of user interface style. Also the teaching of Prinzing'364 would have allowed users to view and interact with various type of Internet resources available on the World Wide Web.

As to claim 2:

Prinzing'206 teaches providing a rich text list class (*e.g., provides a variety of ... classes*) for managing (*e.g., formatting and displaying*) the one or more rich text nodes (*e.g., a hierarchy of text elements*) in the memory structure representation (*e.g., level in hierarchy*); providing a rich text class (*e.g., provides a variety of ... classes*) to create the one or more rich text nodes each representing a unit of rich text and its attributes (*e.g.; provide formatting attributes such as bold, underline, and italic for text*) and instantiating the rich text list class and the rich text class [*col.3, lines 6-53 & col.4, lines 58-66*].

As to claim 3:

Prinzing'206 teaches the representing rich text step includes representing string representations (*e.g., character elements; col.3, line 50 & sequence of characters; col.4, line 65*).

As to claim 4:

Prinzing'206 teaches the string representations comprise, among other things, plain text (*e.g., text 117; col.3, lines 43*).

As to claim 5:

Prinzing'206 teaches providing rich text attributes (*e.g. attributes*), wherein the attributes include, among other things, italicized (*e.g., italic object*) [col.4, lines 58-67].

As to claim 6:

Prinzing'206 teaches providing properties associated with the one or more rich text nodes, the properties comprising, among other things, text (*e.g., text 117; col.3, line 43*).

As to claim 7:

Prinzing'206 teaches the rich text node comprises a table node for defining a table (*e.g., a linear table of attributes; col.4, lines 8-10*).

As to claim 8:

Prinzing'206 does not specifically teach "*the table node includes at least one of a table header node and a table body node, for defining the characteristics and format of the table.*"

Prinzing'364 teaches the table node includes at least one of a table header node and a table body node, for defining the characteristics and format of the table (*see item 416 in fig.4*).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include the feature from Prinzing'364 in the system of Prinzing'206 because it would have provided the capability for generating a text component corresponding to a selectable user interface style and the type of content associated with the text component.

As to claim 9:

Prinzing'206 does not specifically teach "*one or more heading cell nodes, each heading cell node defining another rich text node.*"

Prinzing'364 teaches one or more heading cell nodes, each heading cell node defining another rich text node (*col. 7, lines 22-34*).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include the feature from Prinzing'364 in the system of Prinzing'206 because it would have provided the capability for generating a text component corresponding to a selectable user interface style and the type of content associated with the text component.

As to claim 10:

Prinzing'206 does not specifically teach "*one or more table row nodes for defining an individual row within the table.*"

Prinzing'364 teaches one or more table row nodes for defining an individual row within the table (*col. 7, lines 22-34*).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include the feature from Prinzing'364 in the system of Prinzing'206 because it would have provided the capability for generating a text component corresponding to a selectable user interface style and the type of content associated with the text component.

As to claim 11:

Prinzing'206 does not specifically teach “*one or more row cell nodes for defining rich text in a cell in the individual row, each of the one or more row cell nodes defining another rich text node.*”

Prinzing'364 teaches one or more row cell nodes for defining rich text in a cell in the individual row, each of the one or more row cell nodes defining another rich text node (*col. 7, lines 22-34*).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include the feature from Prinzing'364 in the system of Prinzing'206 because it would have provided the capability for generating a text component corresponding to a selectable user interface style and the type of content associated with the text component.

As to claim 12:

Prinzing'206 teaches providing well-formed segments of text to a current rich text node of the one or more rich text nodes from a rich text list node (*e.g., divide text in document into a hierarchical arrangement of text elements; see item 502 in fig.5*); parsing the well-formed segments of text (*col.2, lines 31-54*); setting attributes in the current rich text node, the attributes including, among other things, bold (*e.g., formatting attributes such as bold; col.4, lines 58-67*).

Prinzing'206 does not specifically teach “*assigning unparsed segments of text to the current rich text node's text attribute; and resolving the current rich text node's text attribute by extracting tag information.*”

Prinzing'364 teaches assigning unparsed segments of text to the current rich text node's text attribute; and resolving the current rich text node's text attribute by extracting tag information (*col.10, lines 6-34 & table 3*).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include the feature from Prinzing'364 in the system of Prinzing'206 because it would have provided the capability for generating a text component corresponding to a selectable user interface style and the type of content associated with the text component.

As to claim 13:

Prinzing'206 does not specifically teach "*suppressing certain tags associated with some the unparsed segments by changing starting and ending tags to substitution strings; checking whether the starting and ending tags are in proper order and eliminating pairs of the starting and the ending tags that have null content; converting some of the substitution strings to original values; and reconstituting the well-formed segments of text into one string when pairs of starting and end tags are eliminated.*"

Prinzing'364 teaches suppressing certain tags associated with some the unparsed segments by changing starting and ending tags to substitution strings; checking whether the starting and ending tags are in proper order and eliminating pairs of the starting and the ending tags that have null content; converting some of the substitution strings to original values; and reconstituting the well-formed segments of text into one string when pairs of starting and end tags are eliminated (*col. 7, lines 10-62 & table 8*).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include the feature from Prinzing'364 in the system of Prinzing'206 because it would have provided the capability for generating a text component corresponding to a selectable user interface style and the type of content associated with the text component.

As to claim 14:

Prinzing'206 does not specifically teach "*restoring table related tags; and breaking the well-formed segments at table tags and organizing the broken segments into a new rich text list node with entries of at least one of vectors and string.*"

Prinzing'364 teaches *restoring table related tags; and breaking the well-formed segments at table tags and organizing the broken segments into a new rich text list node with entries of at least one of vectors and string (col.7, lines 10-62 & table 8).*

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include the feature from Prinzing'364 in the system of Prinzing'206 because it would have provided the capability for generating a text component corresponding to a selectable user interface style and the type of content associated with the text component.

As to claim 15:

Prinzing'206 does not specifically teach "*the text is at least one of hypertext mark-up language and extensible mark-up language.*"

Prinzing'364 teaches the text is, among other things, hypertext mark-up language (e.g., *HTML; col. 6, line 50*).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include the feature from Prinzing'364 in the system of Prinzing'206 because it would have provided the capability for creating different views of text for display on a screen and printing on a printer.

As to claim 19:

Prinzing'206 teaches responding to a request for editing a document containing the rich text; presenting rich text editing controls for editing the document; and accepting changes to the document (*e.g., styles to be added or removed modularly by organizing the text; col.2, lines 44-47*) using one or more classes including a rich text class and a rich text list class for editing the document (*col.3, lines 6-53 & col.4, lines 58-66*).

As to claim 20:

Prinzing'206 teaches the accepting changes step includes accepting changes, among other things, text (*e.g., styles to be added or removed modularly by organizing the text; col.2, lines 44-47*).

As to claim 43:

Prinzing'206 teaches an apparatus for providing a means for representing and managing rich text for used by applications (*see the Abstract*), the apparatus comprising:

- a component representing rich text in a memory structure representation (*e.g., text 117 may be organized as a hierarchy of text elements including a document element, paragraphs elements, and character elements; col.3, lines 43-53*);
- a component providing one or more classes (*e.g., provides a variety of ... classes*) for use by the application (*e.g., accessible to GUI application*) to create the memory structure representation (*e.g., storing formatting information and attributes of the text elements and view objects 121 for formatting and displaying text on display device 105*) [see the discussion beginning at col.3, line 43]; wherein the one or more classes includes, a rich text list class (*e.g., organizes the text in the document into a hierarchical arrangement of text elements*) for managing one or more rich text nodes (*e.g., formatting a text element*) and a rich text class to create one or more rich text (*e.g., Each style object may include the formatting information for different levels of the text element hierarchy. Accordingly, a document style object is associated with the document element, paragraph style objects are associated with the paragraph elements, and character style objects are associated with the character elements. Next, the method associates style objects with the text elements*) each representing a unit (*e.g., a document element, a paragraph element, and character elements*) of a rich text and its attributes (*e.g., attributes*) [col.5, lines 43-62].

Prinzing'206 does not specifically teach the use of Web based applications and browsers.

Prinzing'364 teaches the use of Web based applications and browsers (*e.g., a hypertext markup language (HTML) editor kit 212...for editing...HTML source code... GUI application including Java source code 222, HTML source code 224; see col.4, line 46-col.5, line 16; see also col.11, lines 31-54*).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include the feature from Prinzing'364 in the system of Prinzing'206 because it would have provided an improved method for customizing text components that edit a particular type of text and have a particular type of user interface style. Also the teaching of Prinzing'364 would have allowed users to view and interact with various type of Internet resources available on the World Wide Web.

As to claim 44:

Prinzing'206 teaches a component instantiating the rich text list class and the rich text class; and a component editing rich text in a document using the rich text class (*col.3, lines 6-53 & col.4, lines 58-66*).

As to claim 45:

Prinzing'206 teaches the component for representing rich text includes representing a string (*e.g., character elements; col.3, line 50 & sequence of characters; col.4, line 65*), the string including, among other things, plain text (*e.g., text 117; col.3, lines 43*).

As to claim 47:

Prinzing'206 teaches a component (*e.g. memory 102*) representing rich text in a memory structure representation (*e.g., text 117 may be organized as a hierarchy of text elements including a document element, paragraphs elements, and character elements; col.3, lines 43-53*) and the component (*e.g., a develop kit 'DK' 126*) providing one or more classes (*e.g., provides a variety of ... classes*) for use by the application (*e.g., accessible to GUI application*) and is contained, among other things, a library (*e.g. variety of libraries*) [col.3, lines 43-53]. Refer to claim 43 for rejection of "Web based applications and browsers".

As to claims 48-50:

They are directed to a computer program product for implementing the method of claims 1-3, and are similarly rejected under the same rationale.

As to claim 24:

It is directed to a method for presenting the apparatus of claim 43, and is similarly rejected under the same rationale. Additionally, Prinzing'206 teaches providing well-formed segments of text to the one or more current rich text nodes (*e.g., divide text in document into a hierarchical arrangement of text elements; see item 502 in fig.5*) from a rich text list node to initialize the current rich text nodes for representing rich text in a document (*e.g., provides style objects having information used to describe attributes and*

information for formatting a text element (step 504). Each style object may include the formatting information for different levels of the text element hierarchy; col.5, lines 50-55).

As to claim 25:

Prinzing'206 teaches instantiating the rich text list class and the rich text class (*col.3, lines 6-53 & col.4, lines 58-66*); and editing the rich text in the document using the rich text nodes created by the rich text class (*col.3, lines 6-31*).

As to claim 26:

Prinzing'206 teaches the representing rich text step includes representing string representations (*e.g., character elements; col.3, line 50 & sequence of characters; col.4, line 65 & fig.2*), the string representations including, among other things, plain text (*e.g., text 117; col.3, lines 43*).

As to claim 27:

Prinzing'206 teaches the rich text includes attributes (*e.g., attributes*), among other things, underlined (*e.g., underline*)[*col.4, lines 58-67*].

As to claim 28:

Prinzing'206 teaches the one or more rich text node includes properties (*e.g. attributes*), the properties comprising, among other things, text (*e.g., text elements*)[*col.4, lines 1-4*].

As to claim 29:

Prinzing'206 does not specifically teach "*a table node for defining a table and the table node includes at least one of a table header node and a table body node, for defining the characteristics and format of the table.*"

Prinzing'364 teaches a table node for defining a table and the table node includes at least one of a table header node and a table body node, for defining the characteristics and format of the table (*col.7, lines 10-34*).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include the feature from Prinzing'364 in the system of Prinzing'206 because it would have provided the capability for generating a text component corresponding to a selectable user interface style and the type of content associated with the text component.

As to claim 30:

Prinzing'206 does not specifically teach "*one or more heading cell nodes, each heading cell node defining another rich text node, and wherein the table body node comprises one or more table row nodes for defining an individual row within the table.*"

Prinzing'364 teaches one or more heading cell nodes, each heading cell node defining another rich text node, and wherein the table body node comprises one or more table row nodes for defining an individual row within the table (*col. 7, lines 10-34*).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include the feature from Prinzing'364 in the system of Prinzing'206 because it would have provided the capability for generating a text component corresponding to a selectable user interface style and the type of content associated with the text component.

As to claim 31:

Prinzing'206 does not teach “*one or more row cell nodes for defining rich text in a cell in the individual row, each of the one or more row cell nodes defining another rich text node.*”

Prinzing'364 teaches one or more row cell nodes for defining rich text in a cell in the individual row, each of the one or more row cell nodes defining another rich text node (*col. 7, lines 10-34*).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include the feature from Prinzing'364 in the system of Prinzing'206 because it would have provided the capability for generating a text component corresponding to a selectable user interface style and the type of content associated with the text component.

As to claims 33 and 34:

They include the same limitations as in claims 13 and 14, respectively, and are similarly rejected under the same rationale.

As to claim 35:

It includes the same limitations as in claim 12, and is similarly rejected under the same rationale.

3. Claims 21-23, 32, and 46 are rejected under 35 U.S.C. 103(a) as being unpatentable over **Prinzing** (hereinafter Prinzing'206) in view of **Prinzing** (hereinafter Prinzing'364) as applied to claims 1, 19, 24, and 43 above and further in view of **Domini et al.** (U.S. 6,085,206, issued 07/04/2000).

As to claim 21:

The combination of Prinzing'206 and Prinzing'364 does not specifically teach

“responding to a spell checking request; presenting a spell check panel that displays spelling alternatives to a misspelled word associated with the one or more rich text nodes; and accepting a spelling substitution.”

Domini teaches responding to a spell checking request; presenting a spell check panel that displays spelling alternatives to a misspelled word associated with the one or more

rich text nodes; and accepting a spelling substitution (*col.10, line 1-col.11, line 67 & col.13, line 10-col.14, line 5*).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include the feature from Domini in the system of Prinzing'206 as modified by Prinzing'364 because it would have provided the capability for verifying the accuracy of the grammatical composition of a sentence and the spelling of words within the sentence in an electronic document (*see Domini; Abstract*).

As to claim 22:

The combination of Prinzing'206 and Prinzing'364 does not specifically teach "*the responding to a spell checking request step includes searching a spelling dictionary to locate one or more words for presentation in the spell check panel.*"

Domini teaches the responding to a spell checking request step includes searching a spelling dictionary to locate one or more words for presentation in the spell check panel (*col.20, lines 9-34*).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include the feature from Domini in the system of Prinzing'206 as modified by Prinzing'364 because it would have provided the capability for verifying the accuracy

Art Unit: 2176

of the grammatical composition of a sentence and the spelling of words within the sentence in an electronic document (*see Domini; Abstract*).

As to claim 23:

The combination of Prinzing'206 and Prinzing'364 does not specifically teach "*the one or more words in the dictionary each have one or more associated signatures to aid in locating a match for the misspelled word.*"

Domini teaches the one or more words in the dictionary each have one or more associated signatures to aid in locating a match for the misspelled word (*col.17, lines 37-67*).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include the feature from Domini in the system of Prinzing'206 as modified by Prinzing'364 because it would have provided the capability for verifying the accuracy of the grammatical composition of a sentence and the spelling of words within the sentence in an electronic document (*see Domini; Abstract*).

As to claim 32:

The combination of Prinzing'206 and Prinzing'364 does not specifically teach "*providing a spell checker class for use by the applications for locating replacement words in the document having rich text.*"

Domini teaches providing a spell checker class for use by the applications for locating replacement words in the document having rich text (*col.9, lines 6-67 & col.11, line 61-col.12, line 58*).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include the feature from Domini in the system of Prinzing'206 as modified by Prinzing'364 because it would have provided the capability for verifying the accuracy of the grammatical composition of a sentence and the spelling of words within the sentence in an electronic document (*see Domini; Abstract*).

As to claim 46:

The combination of Prinzing'206 and Prinzing'364 does not teach "*a component for providing spell checking using the memory structure representation.*"

Domini teaches a component for providing spell checking using the memory structure representation (*col.11, lines 9-67*).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to include the feature from Domini in the system of Prinzing'206 as modified by Prinzing'364 because it would have provided the capability for verifying the accuracy of the grammatical composition of a sentence and the spelling of words within the sentence in an electronic document (*see Domini; Abstract*).

Indication of Allowable Subject Matter

4. Claims 16-18 are allowed.

Claims 36-38 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Response to Arguments

5. Applicant's arguments filed 07/05/2006 have been fully considered but they are not persuasive.

Regarding to independent claims 1, 24, and 48: Applicant argues in substance that Prinzing'206 does not teach Web based applications and browsers [Remarks, pages 19, 21].

In response, independent claims 1, 24, and 48 (as previously presented) did not require Web based applications and browsers. Prinzing'364 is combined with Prinzing'206 to teach the added limitations "Web based applications and browsers" (*e.g., a hypertext markup language (HTML) editor kit 212...for editing...HTML source code... GUI*

application including Java source code 222, HTML source code 224; see col.4, line 46-col.5, line 16; see also col.11, lines 31-54).

Regarding to independent claim 43: Applicant argues in substance that Prinzing'364 does not teach the use of Web based applications and browsers [Remarks, pages 20-21].

In response, Prinzing'364 teaches Web based applications and browsers (*e.g., a hypertext markup language (HTML) editor kit 212...for editing...HTML source code... GUI application including Java source code 222, HTML source code 224; see col.4, line 46-col.5, line 16; see also col.11, lines 31-54*). Applications created by using Java source code and HTML source code are "Web based applications". Inherently, Web browsers (*e.g., Microsoft Internet Explorer and Netscape Navigator*) are two common software programs used to locate and display Web pages (*e.g., HTML documents*).

Conclusion

6. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Fries U.S. Pat. No. 6,601,059

Issued: Jul. 29, 2003

7. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **SIX MONTHS** from the mailing date of this final action.

Contact information

8. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Maikhanh Nguyen whose telephone number is (571) 272-4093. The examiner can normally be reached on Monday - Friday from 9:00am – 5:30 pm. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Heather Herndon can be reached at (571) 272-4136.

The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Art Unit: 2176

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Any response to this action should be mailed to:

Commissioner for patents

P O Box 1450

Alexandria, VA 22313-1450

MN

William L. Bashore
WILLIAM BASHORE
PRIMARY EXAMINER